

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2000-82014
(P2000-82014A)

(43) 公開日 平成12年3月21日 (2000.3.21)

(51) Int.Cl. ⁷	識別記号	F I	テマコード [*] (参考)
G 0 6 F 12/16	3 4 0	G 0 6 F 12/16	3 4 0 Q 5 B 0 1 8
3/06	3 0 1	3/06	3 0 1 Z 5 B 0 6 5

審査請求 未請求 請求項の数 9 F D (全 9 頁)

(21) 出願番号 特願平10-265818

(22) 出願日 平成10年9月4日 (1998.9.4)

(71) 出願人 000001007

キヤノン株式会社

東京都大田区下丸子3丁目30番2号

(72) 発明者 春日 信之

東京都大田区下丸子3丁目30番2号 キヤ
ノン株式会社内

(74) 代理人 100081880

弁理士 渡部 敏彦

Fターム (参考) 5B018 GA04 HA11 KA14 MA01 QA05
QA11

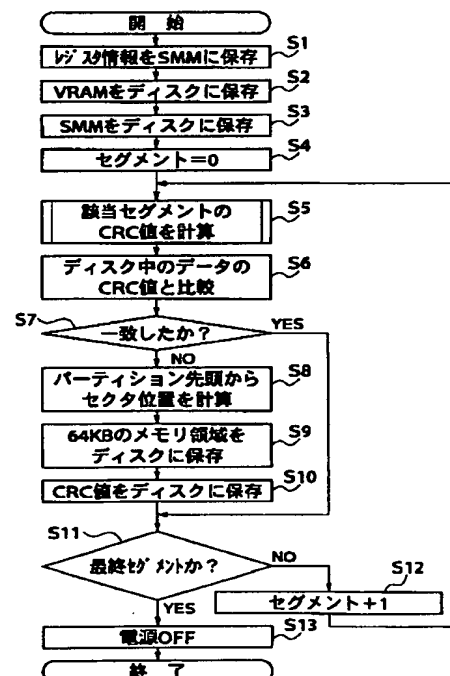
5B065 BA01 EA02 EA23 EA34 ZA14

(54) 【発明の名称】 情報処理装置、情報処理方法、及び記憶媒体

(57) 【要約】

【課題】 ハイバネーション機能によりメモリ内容をハードディスク装置に書き込む際に要する時間の短縮を図る。

【解決手段】 ハイバネーション処理において、メモリ内容をハードディスク装置に保存する際に、ブロック単位にメモリの誤り検出符号 (CRC値など) を求め (S5)、それをハードディスク装置に既に保存されている誤り検出符号 (CRC値など) と比較し (S6)、それが異なる場合にのみ、実際のメモリ内容をハードディスク装置に保存する (S7, S8~S10)。



【特許請求の範囲】

【請求項 1】 ハイパネーションの機能を備えた情報処理装置において、

メモリ内の保存すべきデータの誤り検出符号を算出する算出手段と、

前記メモリのデータを外部記憶装置に最初保存する際に、ブロック単位に前記メモリのデータと共に前記算出手段により算出された誤り検出符号を前記外部記憶装置に保存する保存手段と、

次回以降にメモリデータを前記外部記憶装置に保存する時に、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出手段が今回算出した誤り検出符号とを比較する比較手段と、

前記比較手段による比較の結果、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出手段が今回算出した誤り検出符号とが一致していなかった場合にのみ、前記メモリデータを前記外部記憶装置に保存する保存制御手段とを有することを特徴とする情報処理装置。

【請求項 2】 前記情報処理装置が再起動されたことを検出する検出手段と、

前記検出手段により再起動が検出された場合に、前記外部記憶装置に既に保存されている誤り検出符号を消去する消去手段とをさらに有することを特徴とする請求項 1 記載の情報処理装置。

【請求項 3】 前記外部記憶装置内の破損部分を検出する破損部分検出手段と、

前記破損部分検出手段により破損部分が検出されたときに、前記外部記憶装置の中に代替記憶領域を割り当てる代替領域割当手段と、

前記代替領域割当手段により割り当てられた代替記憶領域にメモリデータを保存する破損時保存手段とをさらに有することを特徴とする請求項 1 または請求項 2 記載の情報処理装置。

【請求項 4】 ハイパネーションの機能を備えた情報処理装置に適用される情報処理方法において、メモリ内の保存すべきデータの誤り検出符号を算出する算出ステップと、

前記メモリのデータを外部記憶装置に最初保存する際に、ブロック単位に前記メモリのデータと共に前記算出ステップにより算出された誤り検出符号を前記外部記憶装置に保存する保存ステップと、

次回以降にメモリデータを前記外部記憶装置に保存する時に、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出ステップで今回算出した誤り検出符号とを比較する比較ステップと、

前記比較ステップによる比較の結果、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出ステップで今回算出した誤り検出符号とが一致していなかった場合にのみ、前記メモリデータを前記外部記憶装置に保存する保存制御ステップとを有することを特徴とする情

報処理方法。

【請求項 5】 前記情報処理装置が再起動されたことを検出する検出ステップと、

前記検出ステップにより再起動が検出された場合に、前記外部記憶装置に既に保存されている誤り検出符号を消去する消去ステップとをさらに有することを特徴とする請求項 4 記載の情報処理方法。

【請求項 6】 前記外部記憶装置内の破損部分を検出する破損部分検出ステップと、

前記破損部分検出ステップにより破損部分が検出されたときに、前記外部記憶装置の中に代替記憶領域を割り当てる代替領域割当ステップと、

前記代替領域割当ステップにより割り当てられた代替記憶領域にメモリデータを保存する破損時保存ステップとをさらに有することを特徴とする請求項 4 または請求項 5 記載の情報処理方法。

【請求項 7】 ハイパネーションの機能を備えた情報処理装置に適用される情報処理方法をプログラムとして記憶した、コンピュータにより読み出し可能な記憶媒体において、

前記情報処理方法が、

メモリ内の保存すべきデータの誤り検出符号を算出する算出ステップと、

前記メモリのデータを外部記憶装置に最初保存する際に、ブロック単位に前記メモリのデータと共に前記算出ステップにより算出された誤り検出符号を前記外部記憶装置に保存する保存ステップと、

次回以降にメモリデータを前記外部記憶装置に保存する時に、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出ステップで今回算出した誤り検出符号とを比較する比較ステップと、

前記比較ステップによる比較の結果、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出ステップで今回算出した誤り検出符号とが一致していなかった場合にのみ、前記メモリデータを前記外部記憶装置に保存する保存制御ステップとを有することを特徴とする記憶媒体。

【請求項 8】 前記情報処理方法が、

前記情報処理装置が再起動されたことを検出する検出ステップと、

前記検出ステップにより再起動が検出された場合に、前記外部記憶装置に既に保存されている誤り検出符号を消去する消去ステップとをさらに有することを特徴とする請求項 7 記載の記憶媒体。

【請求項 9】 前記情報処理方法が、

前記外部記憶装置内の破損部分を検出する破損部分検出ステップと、

前記破損部分検出ステップにより破損部分が検出されたときに、前記外部記憶装置の中に代替記憶領域を割り当てる代替領域割当ステップと、

前記代替領域割当ステップにより割り当てられた代替記憶領域にメモリデータを保存する破損時保存ステップとをさらに有することを特徴とする請求項7または請求項8記載の記憶媒体。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、情報処理装置、情報処理方法、及び記憶媒体に関し、詳しくは、ハイバネーションの機能を備えた情報処理装置、当該情報処理装置に適用される情報処理方法、及び当該情報処理方法を実行するプログラムを記憶した記憶媒体に関する。

【0002】

【従来の技術】従来、パーソナルコンピュータ等の情報処理装置において、パワーマネジメント機能の1つとして、ハイバネーション（休止状態）機能がある。これは、情報処理装置の電源を切る際に、その時点での情報処理装置の動作状態（レジスタやメモリの記憶内容など）を不揮発性のハードディスク装置にすべて保存してから実際に電源を切り、次に電源が投入された際に元の動作状態を復元できるようにしたものである。

【0003】

【発明が解決しようとする課題】しかしながら、近年、パーソナルコンピュータパソコンに搭載されるメモリの容量が急激に増加しており、少し前まで16MBや32MB程度の容量のメモリが搭載されていたものが、今では、その4～5倍の96MBや128MBのメモリが搭載されることが珍しくなくなってきた。

【0004】ハイバネーション機能では、情報処理装置の電源を切断する前に、メモリ内容のすべてをハードディスク装置に保存する必要があるため、メモリ量の増加に伴い、ハイバネーション機能によりメモリ内容をハードディスク装置に書き込む際や、逆にハードディスク装置から読み出してメモリ内容を復元する際に要する時間が非常に長くなっている。

【0005】本発明はこのような問題点を鑑みてなされたものであって、ハイバネーション機能によりメモリ内容をハードディスク装置に書き込む際に要する時間の短縮を図った情報処理装置、情報処理方法、及び記憶媒体を提供することを目的とする。

【0006】

【課題を解決するための手段】上記目的を達成するために、請求項1記載の発明によれば、ハイバネーションの機能を備えた情報処理装置において、メモリ内の保存すべきデータの誤り検出符号を算出する算出手段と、前記メモリのデータを外部記憶装置に最初保存する際に、ブロック単位に前記メモリのデータと共に前記算出手段により算出された誤り検出符号を前記外部記憶装置に保存する保存手段と、次回以降にメモリデータを前記外部記憶装置に保存する時に、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出手段が今回算出した

誤り検出符号とを比較する比較手段と、前記比較手段による比較の結果、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出手段が今回算出した誤り検出符号とが一致していなかった場合にのみ、前記メモリデータを前記外部記憶装置に保存する保存制御手段とを有することを特徴とする。

【0007】また、請求項4記載の発明によれば、ハイバネーションの機能を備えた情報処理装置に適用される情報処理方法において、メモリ内の保存すべきデータの誤り検出符号を算出する算出ステップと、前記メモリのデータを外部記憶装置に最初保存する際に、ブロック単位に前記メモリのデータと共に前記算出ステップにより算出された誤り検出符号を前記外部記憶装置に保存する保存ステップと、次回以降にメモリデータを前記外部記憶装置に保存する時に、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出ステップで今回算出した誤り検出符号とを比較する比較ステップと、前記比較ステップによる比較の結果、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出ステップで今回算出した誤り検出符号とが一致していなかった場合にのみ、前記メモリデータを前記外部記憶装置に保存する保存制御ステップとを有することを特徴とする。

【0008】さらに、請求項7記載の発明によれば、ハイバネーションの機能を備えた情報処理装置に適用される情報処理方法をプログラムとして記憶した、コンピュータにより読み出し可能な記憶媒体において、前記情報処理方法が、メモリ内の保存すべきデータの誤り検出符号を算出する算出ステップと、前記メモリのデータを外部記憶装置に最初保存する際に、ブロック単位に前記メモリのデータと共に前記算出ステップにより算出された誤り検出符号を前記外部記憶装置に保存する保存ステップと、次回以降にメモリデータを前記外部記憶装置に保存する時に、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出ステップで今回算出した誤り検出符号とを比較する比較ステップと、前記比較ステップによる比較の結果、前記外部記憶装置に既に保存されている誤り検出符号と、前記算出ステップで今回算出した誤り検出符号とが一致していなかった場合にのみ、前記メモリデータを前記外部記憶装置に保存する保存制御ステップとを有することを特徴とする。

【0009】

【発明の実施の形態】以下、本発明の実施の形態を、図面を参照して説明する。

【0010】（第1の実施の形態）図1は、本発明の第1の実施形態に係る情報処理装置の構成を示すブロック図である。

【0011】図中、1は中央演算装置であるCPU、2はRAM、3はROM、4はハードディスクコントローラ、5はハードディスク装置、6はキーボードコントローラ、7はキーボード、8はディスプレイコントローラ

ラ、9はCRTやLCDパネルなどのディスプレイ装置、10はビデオメモリ、11は電源コントローラ、12は電源を供給するためのACアダプタである。

【0012】図2は、一般的な情報処理装置のメモリ配置をあらわしたブロック図である。図の中央に示すように、通常のメモリは0～1MB超の領域までリニアに配置されている。しかしながら、ビデオメモリはA0000-BFFFFhまでのメモリ空間にバンク構造で配置されており、またシステム・マネージメント・メモリはD0000hからの64KBの空間に配置されている。これらの特殊なメモリは、通常メモリと同じアドレスを共有し、それぞれを切り替えて使用する。

【0013】図3は、ハイバネーション機能を持つ情報処理装置におけるハードディスク装置のパーティション状態の一例を示す図である。例えば、1つのハードディスク装置に、ハイバネーション領域とDOS領域との2つのパーティションがあり、さらにハイバネーション領域は、制御情報とCRC情報とメモリ情報の3つを含んでいる。

【0014】図4は、本発明を適用したハイバネーション機能によりメモリの状態をハードディスク装置に書き込む際におけるBIOS (Basic Input/Output System) の処理の手順を示したフローチャートである。

【0015】BIOSはまず、ステップS1において各種デバイスのレジスタ情報をシステム・マネージメント・メモリに保存する。例えば、ハードディスクコントローラ4の状態をレジスタから読み出して、システム・マネージメント・メモリ上に保存しておく。これらの情報は、レジャーム(復帰)時に再び各デバイスのレジスタに書き戻され、元の状態を復元する。

【0016】次に、ステップS2において、ビデオメモリ10上のデータをすべてハードディスク装置5に待避させる。このデータも、レジャーム時に再びビデオメモリ10に書き戻され、画面を復元する。次に、ステップS3においてシステム・マネージメント・メモリの内容をハードディスク装置5に保存する。

【0017】次にステップS4～ステップS12において、すべてのメモリをハードディスク装置5に保存する。まずステップS4において、メモリをセグメント(64KB)単位で扱うために、セグメントのカウンタを0に初期化する。次に、ステップS5で、該当セグメントにおける64KBのメモリ内容のCRC(巡回冗長符号検査)値を計算する。ステップS5の詳しい内容については図5を参照して後述する。

【0018】次に、ステップS6で、ここで得られたCRC値とハードディスク装置5に既に書き込まれているデータのCRC値とを比較する。CRC値が一致した場合には、ハードディスク装置5に既に書き込まれているデータと現在のメモリの内容とが一致していると考えられるので、ステップS8、ステップS9、ステップS10

におけるハードディスク装置5への書き込み処理をスキップする。これにより、処理を高速化することができる。

【0019】CRC値が一致しなかった場合には、ステップS8で、ハイバネーション情報領域の先頭からセクタ位置を計算し、ステップS9で、ハードディスク装置5の該当位置に該当セグメントのメモリ内容を保存する。次に、ステップS10で、先程計算したCRC値をハードディスク装置5に保存する。この値は、次のハイバネーション実行時に参照されることになる。

【0020】次に、ステップS11で、最終セグメントに対する処理まで完了したかどうかを判定し、完了していなければ、ステップS12で、セグメントカウンタのカウント値に1を加えて、次のセグメントに対してステップS5～ステップS12の処理を繰り返す。完了していれば、すべてのメモリ内容の保存が完了したので、ステップS13で、情報処理装置の電源を切って処理を終了する。

【0021】図5は、図4のステップS5で行われるCRC値を求める処理の一例を示したフローチャート図である。この例では、16ビットのCRC値を求めている。

【0022】まずステップT1において、CRC値の初期値を設定する。またステップT2で、CRC値を計算する対象メモリを特定するためのポインタの値を0に設定して初期化する。

【0023】次に、ステップT3において、先のCRC値と対象メモリの先頭バイトの値との排他的論理和を計算して、計算結果をCRC値に採用する。次にステップT4で、ビットカウンタを0に初期化する。

【0024】まずステップT5においてCRC値の最下位ビットが1かどうかを判定する。0の場合には、現在のCRC値を右に1ビットシフトして、これをCRC値として採用する(T6)。1の場合には、同じく右に1ビットシフトした後に(T7)、現在のCRC値とCRC定数(8408h)との排他的論理和を計算し、これをCRC値として採用する(T8)。このCRC定数は、CCITTのX.25規格で定められた17ビットのCRC多項式値(11021h)を、このアルゴリズムのビット送り方向にあわせて、ビットの並びを逆順にしたものである。

【0025】次にステップT9で、ビットカウンタのカウント値を1つ増やし、ステップT10により、1バイトすなわち8ビット分に相当する回数(8回)、ステップT5～ステップT9の処理を繰り返す。

【0026】次にステップT11でメモリポインタの値を1つ増やす。1セグメントは、64KBすなわち65536バイトなので、ステップT12により、ステップT3～ステップT11の処理をこのバイト数分繰り返す。すべての処理が終了した時に、64KB分のメモリ

のCRC値が得られる。

【0027】（第2の実施形態）第2の実施形態の構成は、第1の実施形態と同じであるので、その説明を省略し、第1の実施形態の構成を流用して、第2の実施形態を説明する。

【0028】図6は、本発明の第2の実施形態においてハイパネーションを実行する際のBIOSの処理手順を示したフローチャートである。このフローチャートは、図4に示すフローチャートと基本的に同じであるので、同一ステップには同一ステップ番号を付してその説明を省略する。

【0029】一般に、情報処理装置ではシステム自体を再起動すると、メモリの内容が一旦すべて消されてしまうため、それまでアプリケーションなどが使用していたメモリの内容は、その直前の状態と大きく異なってしまう。この再起動状態で上記第1の実施形態の処理を実行すると、CRC値が一致する確率が極めて低く、かえってCRC値が誤って一致してしまった場合の危険性が大きい。

【0030】そこで、第2の実施形態では、情報処理装置の起動時（POST実施時）に、ハードディスク装置中のCRC値を一旦すべて消去（0に初期化）する。そして図6のステップS5'の処理を加えて、ハイパネーション実行時にハードディスク装置中のCRC値が0であるかどうかを判定する。もしCRC値が0であるならば、CRC値の比較は行わず、必ず該当セグメントのメモリを保存する。CRC値が0でない場合には、再起動がされていないので、CRC値の比較を行い（S6、S7）、ハイパネーション動作の高速化をはかる。

【0031】（第3の実施形態）上記の第1及び第2の実施形態では、特定セグメントの内容が格納されるハードディスク装置のセクタはかならず固定位置に存在し、その位置はハイパネーション情報格納領域の先頭から計算し得ることを前提とする。しかしながら、ハードディスク装置に破損セクタがあった場合には、セクタの連続性が失われるため、このような先頭からの計算では位置を特定することができなくなる。

【0032】そこで、第3の実施形態では、この不具合を解消するために、破損セクタが発生した場合には、CRC値と同時に代替セクタの位置を記録して、セクタの連続性を確保する。

【0033】図7は、第3の実施形態におけるハードディスク装置内のハイパネーション領域のデータ構造を示す図である。同図において、ハイパネーション領域にはCRC値と同時にセクタ位置が記録される。セクタ位置情報として、すべてのセクタが正常な場合には、すべて0が書かれる。しかし、破損セクタが発生した場合には、対応位置に予備領域中の代替セクタの位置が書き込まれる。すなわち、図7に例示するように、CRC値がYYYYhの対応部分に破損セクタが発生した場合に

は、予備領域中の代替セクタの位置1234hを、セクタ位置情報として書き込んでおく。

【0034】第3の実施形態の構成は、第1の実施形態と同じであるので、その説明を省略し、第1の実施形態の構成を流用して、第3の実施形態を説明する。

【0035】図8、図9は、本発明の第3の実施形態においてハイパネーションを実行する際のBIOSの処理手順を示したフローチャートである。このフローチャートは、図4に示すフローチャートと基本的に同じであるので、同一ステップには同一ステップ番号を付してその説明を省略する。第3の実施形態では、ステップS7'とステップS9a～ステップS9cとが加えられる。

【0036】まずステップS9において、ハードディスク装置5の該当位置に該当セグメントのメモリ内容を書き込んだ後に、ステップS9aにおいて、その書き込みが正常に完了できたかどうかを判定する。ここで書き込みが失敗した場合には、該当セクタになんらかの欠陥が存在すると推察できる。このため、書き込みが失敗した場合には、図7に示すハードディスク装置内の「セクタ位置」欄を参照して、予備領域中の代替セクタの位置を検索し（S9b）、その位置情報をハードディスク装置に記録する（S9c）。そしてステップS9で再度、この代替セクタに対して該当セグメントのメモリ内容を書き込む。書き込みが正常に終了した場合には、ステップS10で、得られたCRC値をハードディスク装置に書き込む。

【0037】また上記の処理が行われて代替セクタが設定された以降のハイパネーション処理においては、ハードディスク装置への書き込みを実行する前に、ステップS7'で、まずハードディスク装置中のセクタ位置の情報を読み出し、それが0以外であった場合には、従来通り、先頭位置から計算で求めたセクタ位置にメモリの内容を保存する（S9）。しかし、セクタ位置情報が0であった場合には、代替セクタが設定されていると判断し、読み出した代替セクタ位置にメモリの内容を保存する（S8、S9）。

【0038】レジューム時にも、同様にハードディスク装置中のセクタ情報を読みながらメモリを復元し、もし代替セクタが設定されていたならば、予備領域中の代替セクタからデータを読み出す。このようにして破損したセクタに対しては代替セクタを割り当てることで、他のセクタの連続性を確保することができる。

【0039】なお、本発明を、複数の機器から構成されるシステムに適用しても、あるいは1つの機器からなる装置に適用してもよい。

【0040】また、前述した各実施形態の機能を実現するソフトウェアのプログラムコードを記憶した記憶媒体を、システムあるいは装置に供給し、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを読み出して実行

することによっても、本発明が達成されることは言うまでもない。

【0041】この場合、記憶媒体から読み出されたプログラムコード自体が、前述の各実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体が本発明を構成することになる。

【0042】プログラムコードを供給するための記憶媒体として、例えば、フロッピーディスク、ハードディスク、光ディスク、光磁気ディスク、CD-ROM、CD-R、磁気テープ、不揮発性のメモリカード、ROMなどを用いることができる。

【0043】また、コンピュータが読み出したプログラムコードを実行することにより、前述した各実施形態の機能が実現されるだけでなく、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOSなどが実際の処理の一部または全部を行い、その処理によって前述した各実施形態の機能が実現される場合も、本発明に含まれることは言うまでもない。

【0044】さらに、記憶媒体から読み出されたプログラムコードが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行い、その処理によって前述した各実施形態の機能が実現される場合も、本発明に含まれることは言うまでもない。

【0045】

【発明の効果】以上詳述したように、請求項1、請求項4または請求項7記載の発明によれば、ハイバネーション処理において、メモリ内容をハードディスク装置に保存する際に、ブロック単位にメモリの誤り検出符号（CRC値など）を求め、それをハードディスク装置に既に保存されている誤り検出符号（CRC値など）と比較し、それが異なる場合にのみ、実際のメモリ内容をハードディスク装置に保存する。これにより、ハイバネーション処理において、メモリ内容をハードディスク装置に書き込む処理を高速化することができ、そうした書き込み処理に要する時間を短縮することができる。

【0046】また、請求項2、請求項5または請求項8記載の発明によれば、再起動時にはハードディスク装置に既に保存されている誤り検出符号を消去する。これにより、誤ったデータが復元されることを防止できる。

【0047】さらに、請求項3、請求項6または請求項9記載の発明によれば、メモリを保存するためのハードディスク装置のハイバネーション領域に破損セクタがあった場合に、代替セクタを割り当てる。これにより、メモリ内容をハードディスク装置に書き込む処理の高速性を維持することができる。

【図面の簡単な説明】

【図1】本発明の第1の実施形態に係る情報処理装置の構成を示すブロック図である。

【図2】一般的な情報処理装置のメモリ配置をあらわしたブロック図である。

【図3】ハイバネーション機能を持つ情報処理装置におけるハードディスク装置のパーティション状態の一例を示す図である。

【図4】本発明を適用したハイバネーション機能によりメモリの状態をハードディスク装置に書き込む際におけるBIOSの処理の手順を示したフローチャートである。

【図5】図4のステップS5で行われるCRC値を求める処理の一例を示したフローチャート図である。

【図6】本発明の第2の実施形態においてハイバネーションを実行する際のBIOSの処理手順を示したフローチャートである。

【図7】第3の実施形態におけるハードディスク装置内のハイバネーション領域のデータ構造を示す図である。

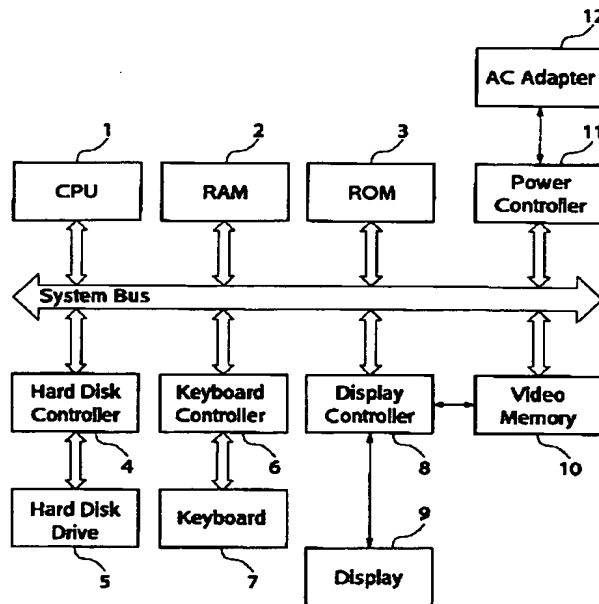
【図8】本発明の第3の実施形態においてハイバネーションを実行する際のBIOSの処理手順を示したフローチャートである。

【図9】図8の続きのフローチャートである。

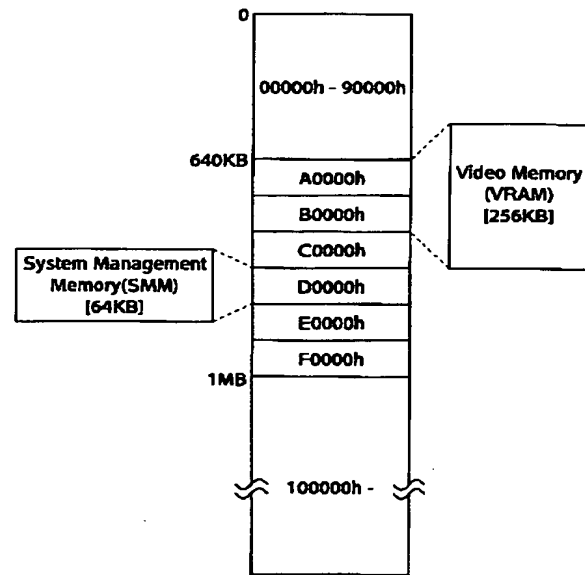
【符号の説明】

- 1 CPU
- 2 RAM
- 3 ROM
- 4 ハードディスクコントローラ (Hard Disk Controller)
- 5 ハードディスク装置 (Hard Disk Drive)
- 6 キーボードコントローラ (Keyboard Controller)
- 7 キーボード (Keyboard)
- 8 ディスプレイコントローラ (Display Controller)
- 9 ディスプレイ装置 (Display)
- 10 ビデオメモリ (Video Memory)
- 11 電源コントローラ (Power Controller)
- 12 ACアダプタ (AC Adapter)

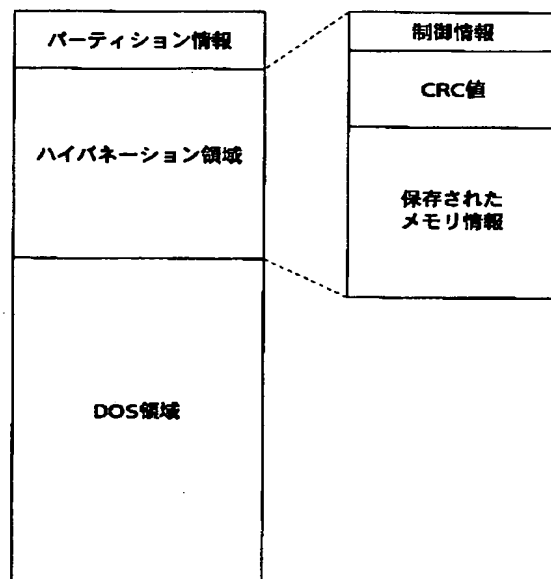
【図1】



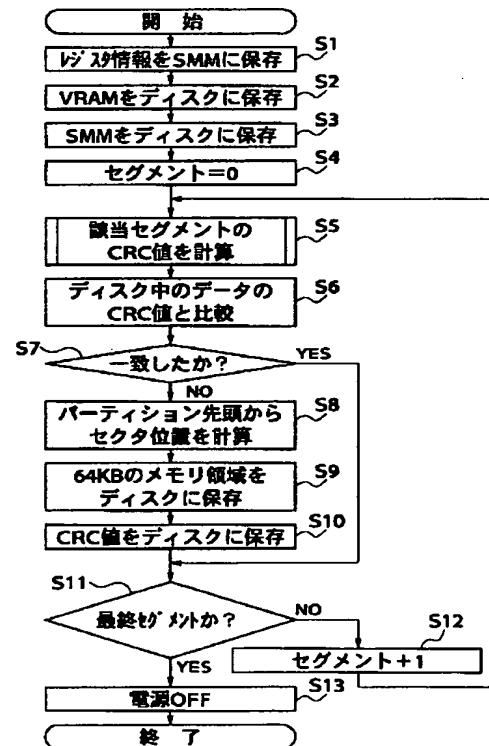
【図2】



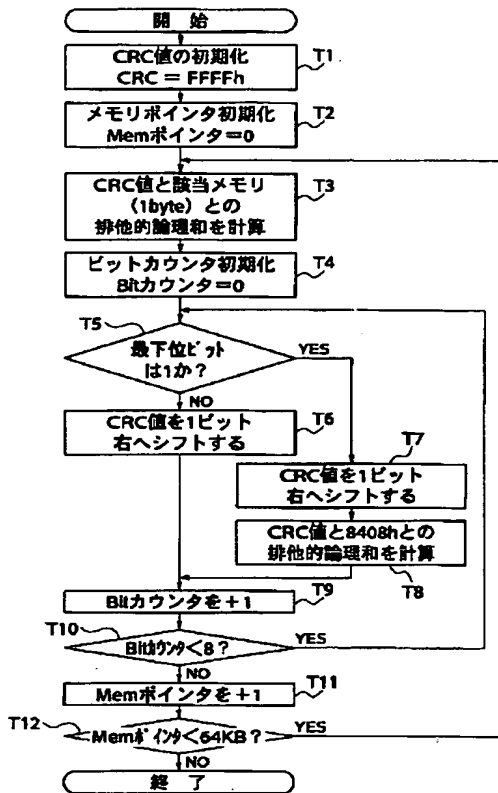
【図3】



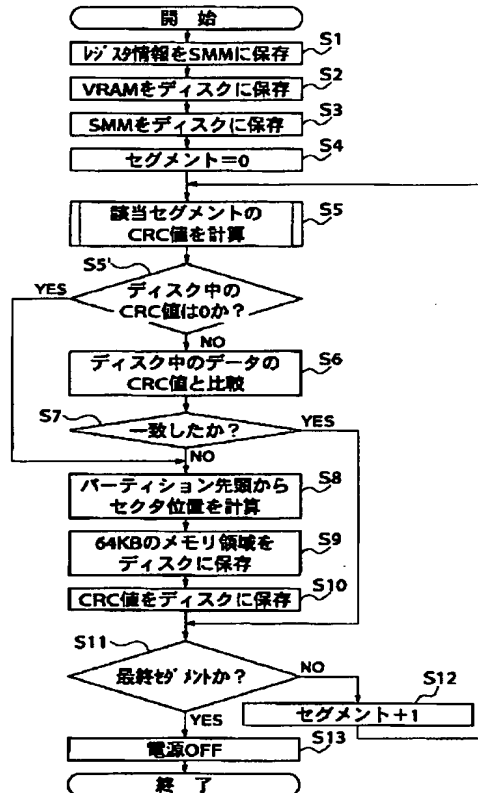
【図4】



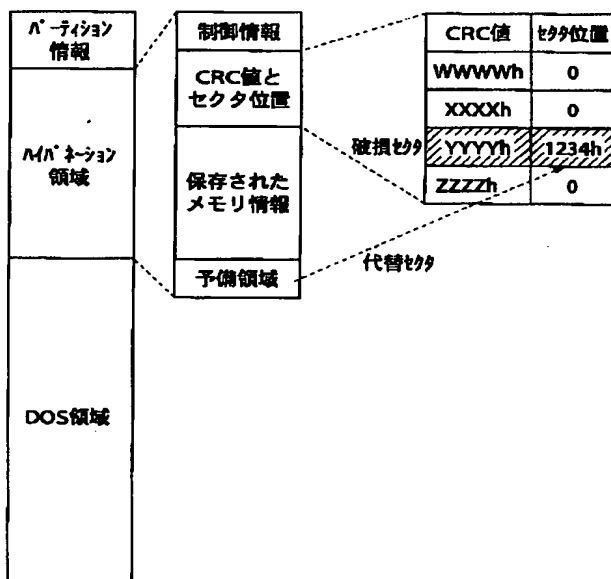
【図5】



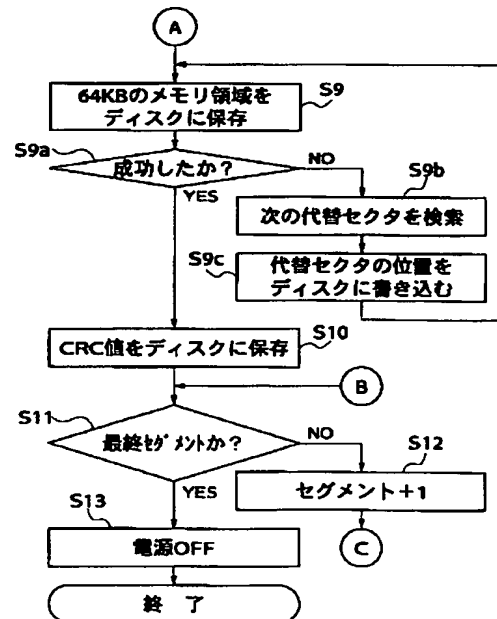
【図6】



【図7】



【図9】



【図8】

